**JS Proc Listing**

**Init Functions:**

loadSysdata() **(called from onLoad)**

calls doclear, startTime

Grabs data from Sysdata for global vars

jumpCursor()**(called from onLoad)**

Puts cursor on CardSwipe input field

startTime()

calls checkTime, grabTime

Displays time on main screen

checkTime()

Adds leading zero to times under 10 in minutes and seconds

grabTime()

Formats time for display on main screen

**Clockin Functions:**

checkClockno(pcValue) **(called from index.html)**

Main routine that checks Card Swipe

Either comes from index.html or via Employee list (CameFromEmpls == "yes" if from list)

Calls clocknoAjax, then parseClockno

clocknoAjax(pcValue)

Checks Clockno vs. ptcards on back end via **"checkclockno.php"**

parseClockno(pcValue)

whichGet = “Empls” from “Clock In Employees”

if clockno bad, throws alert

if clockno good and employee and HasClockIn == ‘yes’, throw up alert:

emplName + " is clocked out. Are you going home?", "Going Home", "Switching Jobs"

if “Going Home”, back to swipe

if clockno good and employee and HasClockIn == ‘no’ OR “Switching Jobs”

calls showbuttons (list of Custs or Conts)

if clockno good and ismgr, and NOT from “Clock In Employees” open main mgr buttons dialog, close swipe div

if clockno good and ismgr, and from “Clock In Employees”

calls showbuttons (list of Custs or Conts)

if clockno good and ismgr, and from “Clock In” (single employee), show alert:

emplName + " is clocked out. Are you finished?", "Finished", "Switching Jobs");

if “Finished”, back to main mgr buttons dialog

if “Switching Jobs,” calls showbuttons (list of Custs or Conts)

getCusts()

Calls custsAjax, then parseCusts

custsAjax()

Gets lists of Customers from server, via **getcusts.php**

parseCusts()

Writes lists of Customers to 3-up buttons, which calls getConts()

getConts()

Calls contsAjax, then parseConts

contsAjax()

Gets lists of Contracts from server, via **getconts.php**

parseConts()

Writes lists of Contracts to 3-up buttons, which calls getTasks()

getTasks()

Calls tasksAjax, then parseTasks

tasksAjax()

Gets lists of Tasks from server, via **gettasks.php**

parseTasks()

Writes lists of Tasks to 3-up buttons, which calls checkClockno()

setupTaskTable()

Builds 3-up table for Employees, Customers, Contracts, or Tasks

setupLetterTable()

Builds 3-up table for Employee Last Name letters so names show alphabetically vertically

doalert()

Shows Alert Box for various clockin functions. One button alert or multi-button if coming back from Clock In.

clearButtons()

Clears all buttons from above tables (values, names) and hides them.

getEmplLetters()

Calls emplLettersAjax, then parseEmplLetters

emplLettersAjax()

Gets lists of Employee Name last Letters from server, via **getEmpls.php?letters=yes**

parseEmplLetters()

Writes lists of Employee Name last Letters to 3-up buttons, and turns them a different color (and enables them) if there employees with last names that start with that letter. Pressing a button calls getEmpls()

getEmpls()

Calls emplsAjax, then parseEmpls

emplsAjax()

Gets lists of Empls from server, via **getEmpls.php**

parseEmpls()

Writes lists of Employees to 3-up buttons, which calls getCusts or get Conts, depending on setting.

showButtons()

Shows lists of buttons for either Customers or Contracts, depending on setting in magic table

clockIn()

Calls clockInAjax, then clockInReturn

clockInAjax()

Clocks user in (if called via Supervisor/Employee Clockin or “Switching Jobs” dialog via **clockin.php**

clockInReturn()

Clocks Employee in if no open time record exists or throws an error if one returned from server.

Clocks Employee out if open time record exists give one of two dialogs depending on whether routine was called by

Employee or Supervisor:

Employee: emplName + " is clocked out. Are you going home?", "Going Home" or "Switching Jobs"

Supervisor: emplName + " is clocked out. Are you finished?", "Finished" or "Switching Jobs"

If Employee and “Going Home,” another alert wishes him/her a nice night.

If Supervisor and “Finished,” returns him/her to main Manager buttons.

If Employee or Supervisor and “Switching Jobs,” calls getCusts or get Conts, depending on setting.

**Manager functions:**

showButtonDlg()

Shows main dialog box for managers. Choices are: “Clock Employee” (which calls “getLetters()”),” “End of Day” (which calls “endDay()”), “Reports” (which calls “mainReport()”), “Settings” (which calls “mainSettings()”), or “Return to Clock” (which closes all Manager dialogs and returns to main swipe window).

endDay()

Brings up dialog that asks for Magic Password.

Calls checkMagic

checkMagic()

Calls getMagic () if password not empty.

getMagic()

Calls magicAjax, then parseMagic

magicAjax()

Validates Magic Password from server, via **checkmagic.php**

parseMagic()

Throws up alert if Bad Password, otherwise throws up dialog offering "Should unmatched entries be cleared?”

If Yes, calls checkUnmatched() with “1” parameter, otherwise calls checkUnmatched() with “2” parameter.

checkUnmatched(pcValue)

If parameter is “1” (“Clear unmatched entries”), it calls “getTimes()” with an empty parameter, otherwise (parameter is ‘2”) it throws up entry form for Entering Cutoff Time. Cutoff time is then checked with “checkCutoff()” before continuing to “getTimes()” with cutoff time entered.

checkCutoff()

Validates that cutoff time is a valid time. If not, stops executing of Submit button, otherwise continues on to “getTimes” with with cutoff time entered.

get Times()

Calls timesAjax, then parseTimes

timesAjax()

Checks for non-Exported records in pttime (either all or based on cutoff time) via **checkTimes.php**

parseTimes()

Gets report from server as to records for export. Will give one of three possibilities:

1. No records (gives a simple Alert with only “Okay” as a choice)
2. Bad records (there are records with either no Timein or no Timeout). Gives a dialog with choice of “View Report” (calls getErrorReport()), “Export” (calls getExport()), or “Cancel.”
3. Okay. Gives a dialog with the choice of “Export” (calls getExport()) or “Cancel.”

In any case, the PHP also sends back the exact SELECT script which is then used later as a parameter to be sent back to the server for use by doexport.php

getErrorReport ()

Calls errorReportAjax, then parseErrorReport

errorReportAjax()

Gets lists of pttime records with missing timein or timeout from server, via **doerrorreport.php**

parseErrorReport ()

Displays above report.

getExport()

Calls exportAjax, then parseExport

exportAjax()

Sends command (either all for cutoff time) to change pttime records to sent = 1 via **doExport.php**

parse Export()

Reports back via alert on how many rows were affected by doExport.php.

doMainReport()

Shows main dialog for choice of reports. Choices are: “Clocked In Only” or “Everything Today,” which both call “getReport()”, or “One Person” which calls intermediary function “getOneReport().”

getOneReport()

Routine which asks for an employee number before then moving on to “getReport()” (or cancelling).

getReport ()

If “One Person” report, checks for blank Employee number and if blank returns false, otherwise calls reportAjax, then parseReport

reportAjax()

Gets report data based on report choice (and employee choice if “One Person”) from server, via **doreport.php**

parseReport()

If no data is returns, throws up alert stating this, otherwise builds a table into which is displays the report data.

mainSettings()

Shows main “Settings” dialog, which in turn calls “getSetting()” based on the choices:

“Edit Employee Card IDs,” “Rounding,” “Program Options” or “Cancel” (which returns user to the main manager dialog).

getSetting()

Routes calls based on Setting choice passed. If “Edit Employee Card IDs,” it routes to getCards(). If “Rounding,” it throws up the main “Rounding” screen. If “Program Options,” it throws up main “Program Options” screen. If the user wishes to change the Password, it takes input from the “existing password” field and in turns calls the “checkPassword()” routine. If that routine returns a “good password,” if opens a “new password” field into which to enter a password. If “Cancel,” it returns user to the main manager dialog.

saveSettings()

Calls saveSettingsAjax, then parseSaveSettings

saveSettingsAjax()

Saves setting from the “Rounding” screen to the server, via **savesettings.php.**

parseSaveSettings()

Returns message alert to the user (“save” or “fail”) and then retrieves modified sysdata information for the global variables of the system.

checkPassword()

Calls passwordAjax, then parsePassword

passwordAjax()

Check for valid password in magic table (for either End of Day or changing password).

parsePassword()

Validates password and either sends “Bad Password” alert of allows continuation (opens “New Password” field if changing the password or allows “End of Day” to continue).

saveOptions()

Calls optionsAjax, then parseOptions

optionsAjax()

Converts the screen data to correct formatted values, which it in turn sends on to the server, via **saveoptions.php.**

parseOptions()

Throws up a “Options Saved” or “Options Save failed” in the Options dialog.

getCards ()

Calls cardsAjax, then parseCards

cardsAjax()

Gets lists of cards available for edit from server, via **getCards.php**

parseCards ()

If no cards available, throws up alert informing user of this. If cards available, throws up a table of cards, with “card number” available for editing, along with displays of employee number, first name, last name, and the “Manager?” checkbox available for editing. The dialog then gives the options of “Save” (which calls “saveCards()”), “Cancel” (which returns user to the main Options dialog), or “Add Manager” which calls “addCard().” The “Manager?” input field also has a validation which calls the “checkCard()” routine. Additionally, each record has a hidden “ID” field which houses the ID field number from the server. If a new record is added on the client side, the ID will be zero, which tells the server side to do an INSERT instead of an UPDATE on a “Save.”

NOTE ON CARD NUMBER: The ptcards file gets rebuilt (for non-managers) every time the data is down and uploaded to Activity Tracker. The card number field itself will be populated in one of three ways, based on the radio button setting on the options screen:

1. The card number is the same as the Employee number.
2. The card number is populated from the “Text3” field of the Employee record in Activity Tracker.
3. The card number is blanked out and waits for input.

checkCard()

If the “Manager?” checkbox is checked, routine enables the first and last name for editing, otherwise it disable those two fields.

addCard()

Adds a row to the top of the cards table, with blanks in all fields except the “Manager?” checkbox, which is checked and not available for editing. The clock number and the two name fields are available for editing. The Employee number is blank and not available for editing.

saveCards()

Routine initially runs a number of checks of the cards. It checks:

Are there are any blank card numbers? If so, return false with alert message on screen.

Are there are any duplicate card numbers? If so, return false with alert message on screen.

Are there are managers with blank first or last names? If so, return false with alert message on screen.

If all tests are passed for the group, then the routine calls saveCardsAjax, then parseSaveCards

saveCardsAjax()

Sends accumulated card data to the server, via **savecards.php.** If the ID number sent to the server is zero, it tells the “Savecards.php” to do an INSERT instead of an UPDATE.

parseSaveCards ()

Displays the message returned from the server (for number of INSERTS and/or UPDATES), and then repopulates the card table.

**Utility Functions**

showvalue()

This is the main “hub” which decides how the various “button screens” get populated. They can be populated with Customers, Contracts, Tasks, Employees, Employee last name letters, or even returns from clocking in or out.

doClose()

This is the main routine that “cleans up” the alert div and exits to the main Manager choice dialog.

doClose2()

This is the main routine that “cleans up” the alert div and is used in conjunction with other calls that come after it.

doClear()

Clears out message boxes, global Employee and Job values and returns user back to either main swipe screen or the manager choice dialog, depending on the routine.

checkNumeric(pcValue)

Checks to make sure the data input in the “Round up” field (timeslice) of the “Rounding” screen is indeed a numeric character.

checkBreakpoint(pcValue)

Checks to make sure “Round Up” field (timeslice) is both a numeric as well as less than the “Breakpoint” field’s value.

okayPassword(pcValue)

Makes sure the “new password” entered is both not blank as well as not less than 5 characters in length.

doinclude()

Main initial setup routine for both the main Manager button dialog as well as the alert div.

**Upload and Download PHP script:**

**Jsontcxfer.php** – Takes upload data from Activity Tracker for Customer, Contract, Task, and Employee tables inserts it into the corresponding tables after clearing out those tables. It also adds Employees to the Card table after clearing that table of all EXCEPT the Managers.

Sends back to Transactions from PTTIME table to Activity Tracker and then copies the PTTIME data to OLD\_PTTIME table and clears out the PTTIME table for all records marked as “Sent.”